

Weekly Report

Period: 26/07/17 - 03/08/17

Reporter: 高翔

1 Last Week

1. MedicalVis

目前的进展是我已经把体绘制绘制出来了，下一步的工作将其移到之前 face tracking 的结果上。体绘制的方法参考：<http://www.lebarba.com/>，其中由于 WebGL 不支持 3D texture，所以只能先把 raw 文件切割成 2D 切片（刚才这个网站提供了切割工具）。虽然还是用 ray casting 的方法做 volume rendering，但数据变成了 2d texture。

目前的效果截图如 Figure 1 所示：

所以现在的难点在于如何将体绘制出来的结果和之前的 face 配准。

2. 新加坡同学的 DATARVis 项目

总体来说，新加坡同学完成了一个非常不错的工作，但是离我预想的 AR 效果还是有一定的差距。他们的系统可以分为三部分：

1. 人脸识别部分。识别出具体是谁
2. 信息展示部分。识别出是谁之后，展示其个人信息
3. 好友关系展示部分。两个视图，一个是力导向图，另一个是直接展示出最近一层的好友。

DATARVis 首先会展示一个主界面（具体参考项目文档），然后通过这个主界面可以进入到三个子界面，第一个子界面，第一个页面是用户提交自己个人信息的界面，第三个页面是展示用户的最近活动，第二个页面就是核心页面，人脸识别。

系统的实现流程图如 Figure 2 所示：

最有技术含量的功能就是人脸识别完之后把个人的信息展示出来，本以为他们会把力导向图和人脸最后结合起来，结果没有做到，不过如果我来改进的话 应该可以。

整个项目所用的技术栈可以参照 Figure 3。

前端文件的 serve 和后端数据库的查询都基于 Flask 这个框架来做，部分数据如人脸照片，人脸数据会提前用 python 脚本生成。前端的代码主要是 tracking.js 用来做人脸追踪，然后通过 canvas 把识别到的人脸截图并传给后端，后端匹配完成后返回该用户的信息。人脸识别用了 dlib 这个现成的库。

目前我觉得可以考虑一下如何把图数据 map 到现在的人脸识别视频流中，或者重新想一种结合的方式。现在新加坡的同学把所有流程跑通了，就看我们接下来怎么组合把其变成一篇 paper。

3. 学习 WebGL

阅读了《交互式计算机图形学》的前三章。

2 Next Week

1. 将 volume rendering 结果先移到之前的 face tracking 画面中。再考虑怎么配准
2. 继续学习 图形学和 WebGL

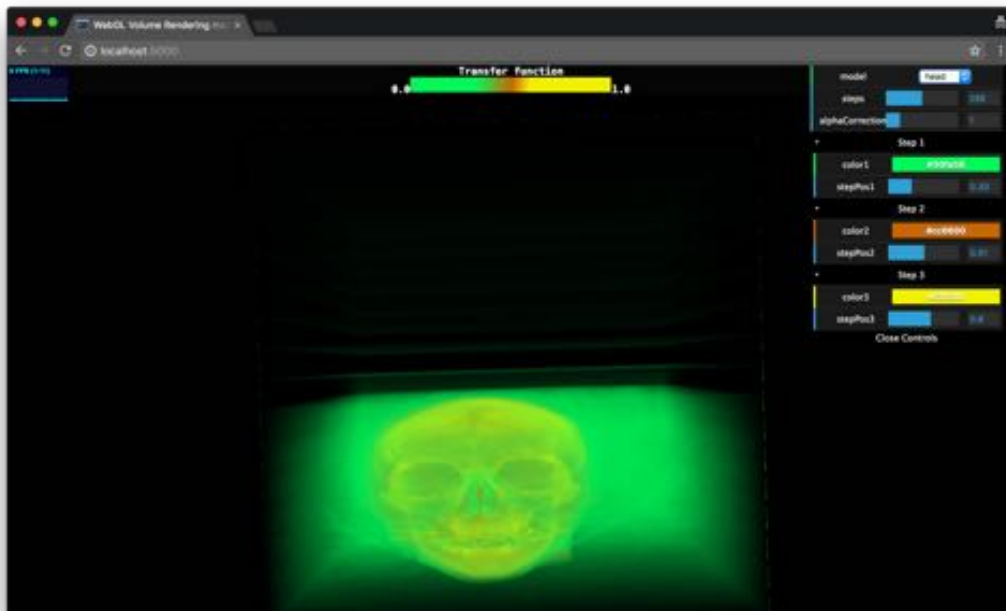
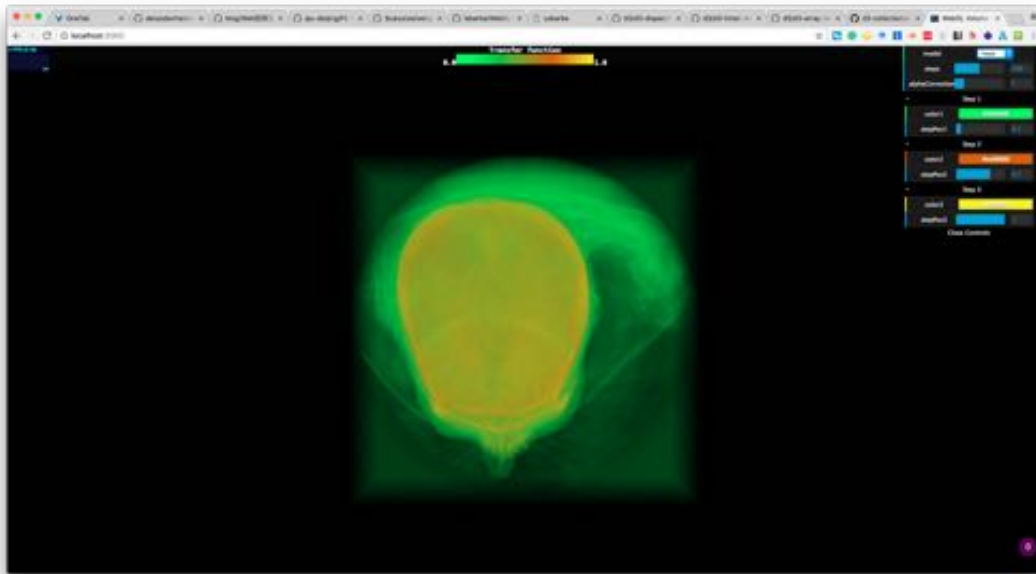


Figure 1: Web Volume Rendering



Figure 2: A flowchart that demonstrates how the entire application works

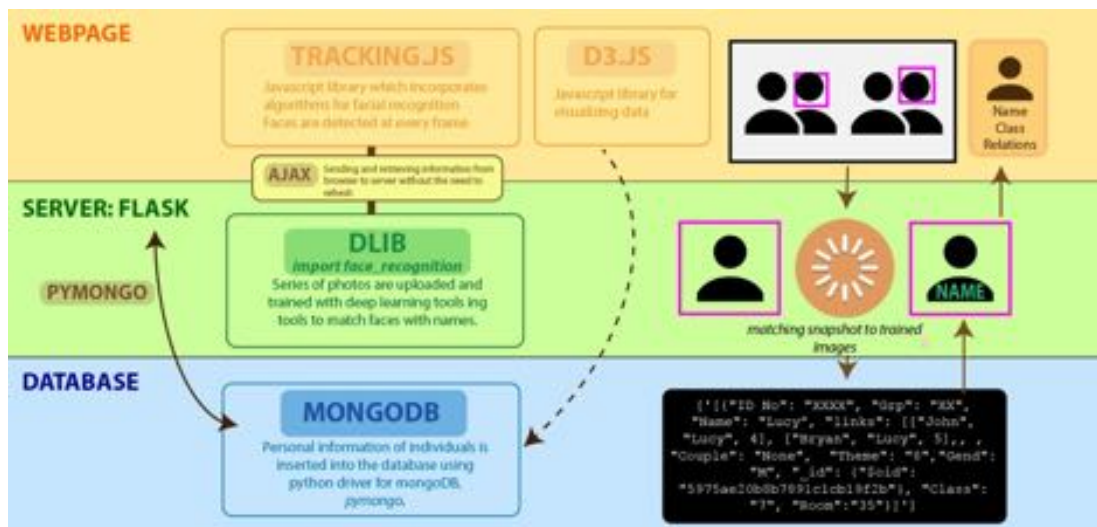


Figure 3: Schematic diagram demonstrating technical aspects of the application